TOWARDS AN AUTOMATED PIPELINE FOR THE TRANSLATION AND OPTIMIZATION OF GEOSPATIAL DATA FOR VIRTUAL ENVIRONMENTS

Ryan McAlinden*[1], Matt Bosack[1], Adrian Macha[1], Esau Vargas[1], John Mann[2], Tim Walker[2], and Julio de la Cruz[3]

[1]Institute for Creative Technologies, University of Southern California
Marina del Rey, CA 90292

[2]Applied Research Associates, Inc.
Orlando, FL 32826

[3]U.S. Army Research, Development, and Engineering Command Simulation and Training Technology Center
Orlando, FL 32826

ABSTRACT

The infusion of commercial game technology into U.S. Army training, simulation, and instructional domains has resulted in more immersive and engaging experiences for Soldiers to hone their skills. However, the influx of such technology comes at a significant cost, specifically in the creation of virtual environments in which these skills are simulated and practiced. Today's typical commercial triple-A game title cost upwards of $40-$60M and four to six years to develop, much of which is spent on producing the digital assets used to populate the scene (models, animations, etc). Additionally, this content is often suited for a custom type of rendering technology, and often cannot be reused without significant manual modification. Unfortunately, the Army has neither the financial or personnel resources available to create such highly immersive, reusable virtual content, nor the time to invest when current operations call for training or simulation data in a matter of hours, not months or years. In this paper, we discuss a research initiative aimed at significantly reducing the time and cost for converting, optimizing, and enhancing existing geospatial data for today's virtual environments. The goal is a completely automated process for ingesting existing military terrain data and outputting a technology-agnostic representation in less than 24 hours.

1. MOTIVATION

The past five years has witnessed a significant increase in the use of commercial game technology adopted for the military training and simulation domains. This technology has the capability to produce highly immersive and detailed 3D environments. However, the construction of these synthetic experiences (the terrain, characters, and animations) often requires significant investment by asset providers (artists, modelers) to create and assemble the virtual landscape, which is often a very arduous, time-consuming, and expensive process. Additionally, as game technology increases support for advanced rendering techniques (such as per-pixel shading, high dynamic range imagery, and multitexturing), the cost for creating these synthetic experiences continues to increase at a rapid pace. The average commercial video game today costs upwards of $20 - $50M to design and produce (BBC, 2005), and a majority of the production staff consists of modelers, animators, and texture artists to create the assets seen by users. However, neither Army nor Academia has such resources available to keep pace with these technical advancements and user expectations, and as a result many of the 3D visual representations seen in today's Army training systems are less than adequate for many types of training and mission rehearsal.

Paradoxically, the U.S. Army has spent millions of dollars developing runtime databases for the same geographic areas, each time targeting different levels of resolution and different data formats to meet individual program requirements. The differences in processing techniques for these databases have caused correlation problems that make interoperability between simulation federates difficult or impossible. Additionally, the terrain database generation systems used to create these databases are tailored to a specific project, often with no attempt to provide a path for reuse by other projects. The runtime databases produced by these systems have particular features and attributes that meet only the requirements of a specific program. Once the databases are produced, additional manual processing is required to make the database usable by runtime systems.

To address these myriad of challenges, the Research, Development, and Engineering Command's (RDECOM) Simulation and Technology Training Center (STTC) has initiated a research effort that is investigating and developing a set of processes and tools for the rapid *conversion*, *manipulation*, *optimization*, and *enhancement* of military geospatial databases for use with the latest

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **DEC 2008**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **Towards An Automated Pipeline For The Translation And Optimization Of Geospatial Data For Virtual Environments**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Institute for Creative Technologies, University of Southern California Marina del Rey, CA 90292**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002187. Proceedings of the Army Science Conference (26th) Held in Orlando, Florida on 1-4 December 2008, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**11**	

commercial game technology. Employing commercial and government off-the-shelf (COTS/GOTS) software, the Military Terrain for Games Pipeline (MTGP) aims to significantly reduce the *time* and *cost* associated with recreating geo-typical and geo-specific environments for the virtual domain. This paper presents the MTGP research effort, including details of the original military geospatial data, how this data is converted/augmented for use in a variety of real-time, game-based environments, and finally results from dataset testing performed to-date.

Though the research goals of the MTGP center around developing automated tools and techniques for reducing the time and cost for creating datasets for an assortment of game platforms, an indirect objective is to begin bridging two disparate communities responsible for database generation: 1) those dealing with strict geo-specificity, correlation, and accuracy issues, and 2) those interested in immersion and aesthetics. Often times, due to project and program requirements, these two communities clash. The correlation community requires data that is consistent and correct, and is often created for constructive simulation systems. The fidelity/aesthetics community requires data that is fully-featured, dense, and realistic of an area of interest. However, as immersive virtual environments continue to infiltrate the DoD training and simulation space, these mutually exclusive communities must partner to form a common set of requirements and methods for representing terrain in both the constructive and [game-based] virtual environments.

2. RELATED WORK

Along with recent advances in hardware and computer-generation imagery (CGI) content creation tools used in movies and games, a heavy push towards procedural generation of virtual urban environments can be clearly observed in the game industry (Introversion, 2008). Currently there are many tools available to assist in automating the production of large urban scenes, ranging in capability from easy-to-use building designers to full cityscape generators. CityEngine is one such tool that uses shape grammars to procedurally generate large cities via randomization of user-specified texturing and architectural details (Procedural Inc., 2008). The result is a realistic environment for a small fraction of the production cost incurred when traditionally using a team of artists and modelers. Although commercial tools such as CityEngine are highly effective in producing realistic scenery, they are primarily geared towards geo-typical content. These tools usually are not capable of ingesting standard military source data (ESRI Shape files, LIDAR, etc), or for producing geo-specific environments for training applications.

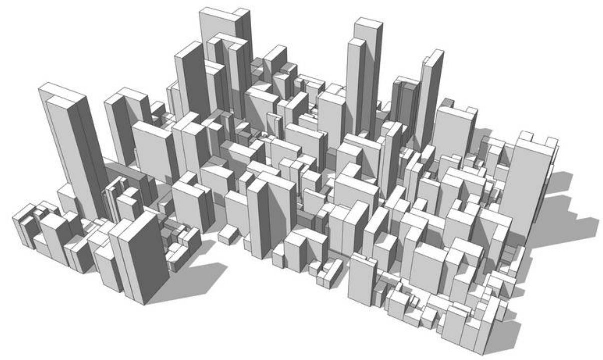

Fig. 1 CityEngine geo-typical cityscape

Even when such tools are adapted for this purpose, they are incapable of producing many non-visual correlated output formats such as compact terrain databases (CTDB) and the OneSAF terrain format (OTF).

In addition to the efforts targeting terrain generation and non-visual correlated output, there exists several initiatives by the game industry to streamline the creation of their virtual environments. Often the solution entails designing and employing a formalized pipeline that dramatically reduces the time and cost associated with creating virtual content. This need has arisen from an industry that must publish a single game title for up to seven different platforms in three major North American markets (Stanford, 2004). As a result, studios must take measures to reduce the risk associated with producing content for a game that may never be successful. Many (if not all) of the Game Developer Conferences (GDCs) in recent memory have included a panel or technical session dedicated to the content pipeline, where leaders from industry assemble to discuss the most effective tools and techniques for streamlining the creation of virtual media. Though the actual implementation of a pipeline by individual studios often centers around custom requirements and technologies (such as a particular 3d authoring environment or game engine), attempts have been made by the broader game community to adopt techniques and standards for streamlining content creation. One such standard is COLLADA, which defines an interchange file format for interactive 3D applications (COLLADA, 2008). Implemented as an open-source XML schema, COLLADA data provides a lossless mechanism for storing and sharing all facets of digital art such as polygonal models, meshes, animations, physics, and programmable shader effects. The standard has been adopted by several leading commercial vendors, and as a result COLLADA data can be imported and exported by several application types. This includes Autodesk's Maya and 3dsMax, LightWave, Softimage/XSI, Houdini,

Blender, and SketchUp, all leading content creation tools used in the game industry.

Unfortunately, despite the recent trend towards interoperability standards such as COLLADA, many current commercial and DoD terrain pipeline capabilities are designed to operate within very narrow requirements and technologies. For example, only a single game engine may be supported for asset export (Gamebryo, Unreal), or a single type of source data (DTED, DEM, LIDAR) for import. However, these restrictions are only partly due to the requirements of the program employing the pipeline. Often times there exists significant disparity between incoming source data, as well as with the rendering techniques employed by different game engines, which makes generalizing a set of pipeline features overly difficult. For example, reusing an asset in the Gamebryo game engine originally developed for the Unreal game engine often requires a significant amount of manual manipulation so it conforms with the underlying Gamebryo renderer. This is the main research thrust of the MTGP.

3. RESEARCH APPROACH

The research approach for generating game-ready visual databases is broken into two sections below. The Rapid Unified Generation of Urban Databases (RUGUD) section discusses how varieties of source data are ingested, manipulated, and correlated to produce a standardized visual representation in COLLADA. The MTGP section details how this COLLADA data is manipulated, optimized, and enhanced to produce a game engine-agnostic visual database for use in a variety of commercial rendering platforms.

RUGUD

The Rapid Unified Generation of Urban Databases (RUGUD) system was developed to address the limitations of current terrain database generation systems (Campbell et al., 2006). Rather than replace existing systems, RUGUD was designed to leverage them, integrating the features of multiple COTS, GOTS and open source tools into a single framework to provide a single processing pipeline (Fig. 2). By supporting multiple tools, RUGUD is able to use best of breed capabilities to automate individual parts of the terrain generation process. From a user's perspective, RUGUD presents a single tool with individual processing capabilities represented as drag and drop components in the pipeline.

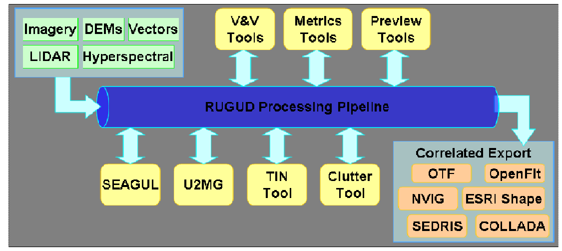

Fig. 2 The RUGUD processing pipeline

RUGUD addresses the problem of database reuse by using a Master Urban Database (MUDB) to store source data. The MUDB is based on a data model that contains a superset of all terrain database content requirements. Individual databases produced for different runtime products pull data from the same database, but only use the data they require. The process of using a master database for all runtime databases ensures that the databases produced are correlated. Road locations, buildings, cultural features, and terrain heights match between databases used for SAFs, image generators, and game engines. One of the components RUGUD uses to eliminate much of the manual processing required in database production is the Urban and Underground Model Generator (U2MG) (Mann and Eifert, 2006). U2MG takes shape file footprints, building heights, and building types as input and automatically generators 3D buildings with interiors as output. RUGUD integrates these buildings with the terrain surface to create a seamless urban database.

Although RUGUD was initially funded as a GOTS solution for the arduous task of constructing large urban terrain databases for military training, it was fundamentally designed as a general-purpose data processing framework based on a plug-in architecture. The framework is centered on the Master Urban Data Model (MUDM), which defines the superset of attribution and feature information required to ultimately convert sets of source data into desired output formats. Plug-ins used for importing, processing, and exporting of different data sets can be written independently and registered with the RUGUD pipeline to expand current capabilities. In this fashion, support for game-related formats (COLLADA, Maya, etc.) are easily implemented by developing a new plug-in. Since the plug-ins all reference the same MUDM attribution and are based on the same source data, RUGUD can correlate the different database outputs to the highest common fidelity shared between formats. The RUGUD GUI provides a drag-and-drop mechanism to create a custom processing pipeline that will include only the source data and output formats desired (Fig. 3).

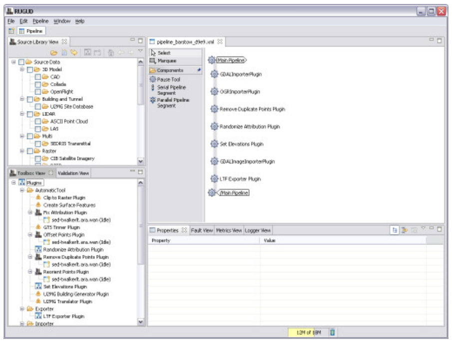

Fig. 3 RUGUD GUI with source data, plugins, and pipeline views

Of course, in order to facilitate the production of urban databases, a tool is needed to simplify the creation of building models as well. U2MG fulfills this need by automating the production of geo-typical buildings with interiors that adhere to geo-specific footprints acquired from source data. U2MG was designed as both a standalone application and a plug-in to the RUGUD framework, capable of directly converting source areals and attribution (height, layout type, number of floors, etc.) to building models with navigable interiors. Recently, many new features were added to improve the geometry of auto-generated buildings. These include support for non-rectangular apertures, exterior columns (rectangular and arc), arched ceilings, window ledges, and interior layout generation based on exterior aperture placement. In standalone mode, U2MG can also be used to generate geo-specific interiors, using a simplified CAD-like interface that requires no traditional modeling expertise. U2MG-generated building models can be exported to multiple visual and SAF formats and integrated into the terrain database via the RUGUD pipeline.

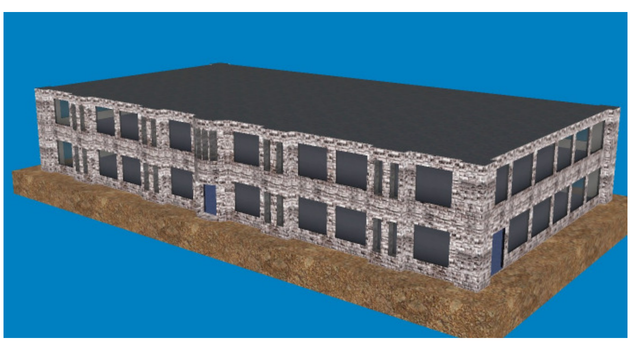

Fig. 4 An auto-generated building in U2MG

The RUGUD and U2MG programs have always focused on easing the task of generating correlated, high-resolution urban terrain databases, but initial users of these tools were far more interested in support for SAF terrain databases (CTDB, OTF, etc.) than in supporting leading-edge correlated visuals. However, since the core intent behind the development of RUGUD has always been to establish a flexible data manipulation framework, it eventually made sense to apply this infrastructure towards production of the more impressive visual databases used in gaming environments. Thus, within the past couple of years, RUGUD and U2MG have undergone efforts to enhance visual output capabilities while maintaining desired SAF correlation wherever possible. Enhancement tasks have included ongoing support for the COLLADA interchange file format, terrain database tiling, proper triangulation of building geometry, more stylistic building facades, and researching/developing export capabilities for modern gaming engines (Half-life 2, Unreal, etc.).

MTGP

The MTGP begins with the ingestion and conversion of correlated COLLADA data produced by RUGUD. Once converted to COLLADA, the source data may be exported directly to a rendering environment that supports the interchange format (such as the Unreal 3 Engine (Epic, 2008)), though the data often requires additional manipulation before final rendering to optimize and enhance the scene. This automated manipulation process is performed inside of Maya, one of the leading content creation tools used in the game industry. The decision to employ Maya was threefold. First, because of its ubiquity throughout the game industry, many of the leading game engine technologies provide exporters directly from Maya to their internal formats for final render (Unreal, Gamebryo, Crysis, Ogre, Torque). Second, Maya provides a game engine-agnostic format for representing virtual terrain such that content created in Maya can be reused across several engines (though there are several caveats to this, discussed in detail below). Lastly, Maya includes a powerful, robust programming language (MELScript) for procedurally manipulating parts of the scene. This is a critical feature as it allows us to algorithmically alter the geometry, textures, and lighting components of the data. Additionally, because the original source data imported into RUGUD is often procedurally generated, algorithmically manipulating procedurally-created data proves to be much simpler than interpreting the creative steps taken by an artist. These algorithmic operations includes operations such as duplicate edge removal, vertex-welding, normal reassignment, and refactoring of the UV layout.

The result of our efforts is an automated process (pipeline) for producing immersive 3-D environments. The first step for automating the pipeline involves the use of MELScripts to import the COLLADA data into a Maya scene. Next, we are able to employ the use of MELScripts and the Maya API to automate the optimization and enhancement of the data. This is the most critical part of the pipeline automation, because it transforms the source data into a highly detailed and complex virtual environment. An artist or modeler

typically bears this manually intensive job, and the tasks involved can consume significant time and resources. Lastly, the optimized data is then exported to a supported game engine (see Fig. 5).

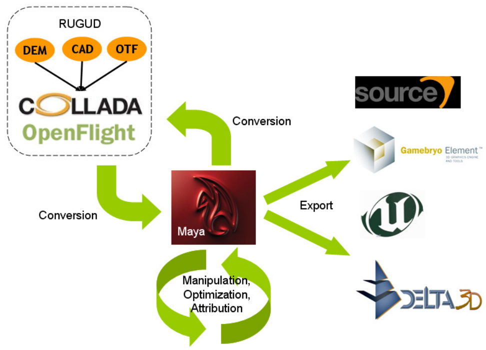

Fig. 5 Military Terrain for Games Pipeline

As mentioned, optimizing the data is paramount, and in order to develop algorithms that, in a timely and cost-effective manner, would output results that rival the fidelity of what an artist would produce, the MTGP pipeline utilizes the collaborative efforts of a research programmer and an artist to determine which optimization and enhancement steps can be automated. This collaborative union has identified three major individual areas of fully automated enhancement and improvement:

1. Scene cleanup: removing unnecessary geometry such as duplicated edges or polygons, removing invisible objects, and merging duplicate material shaders.

2. Terrain texturing: applying high-resolution textures such as ground textures, roads, grass, etc. to the terrain skin.

3. Building texturing: applying assorted high-resolution geo-typical textures to single and multi-elevation structures in the scene.

Optimization tools were then developed to automate these three typically arduous tasks. To begin optimizing the scene, it is imperative to remove all meshes and objects that are not necessary in order to save rendering resources (i.e. scene cleanup). In this step, we check that level of detail (LOD) functionality is properly specified within Maya. The source data can contain up to hundreds of buildings, each with individual levels of detail, producing a immensely polygon-intensive scene. Furthermore, buildings can have duplicate edges and polygons, or unnecessary duplicate material shaders. To address this, MELScripts have been created that recognize and correct these issues in order to properly optimize the scene in order to achieve better frame rates during real-time render.

Terrain texturing is the next procedural enhancement, though this can often be quite difficult as the source data typically lacks any usable texture information or is missing textures altogether. In such cases, it becomes necessary to retexture the ground plane procedurally. Performing this task entails generating custom UV maps. We developed a plug-in for Maya to facilitate this process instead of relying solely on MELScripts. This plug-in utilizes the Maya API to maximize performance since the algorithms to compute new UV maps are computationally intensive. The algorithms to generate custom UV maps and textures consist of three parts:

1. Analyzing the Terrain
    - Distinguish between different types of terrain (Wide roads, narrow roads, square fields, rectangular fields)
    - Search for outer edges for each type of terrain
    - Find parallel edges for streets
    - Compute minimal bounding parallelograms for rectangle shapes
2. Assigning new materials depending on the type of the terrain
3. Computing three different types of UV maps for streets, square shapes and rectangle shapes

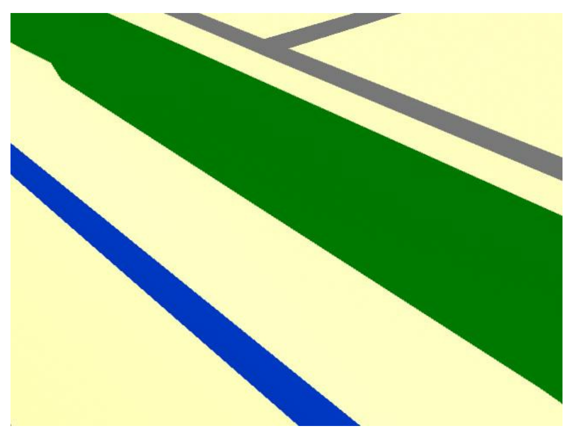

Fig. 6 Original COLLADA data

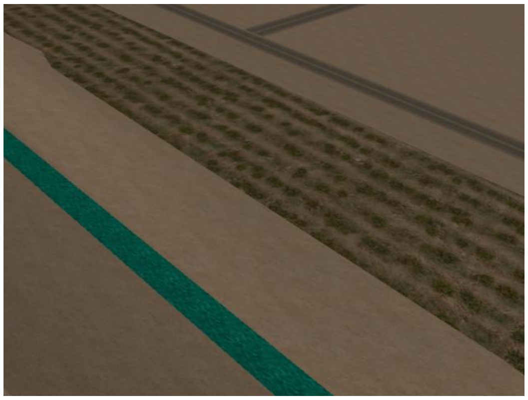

Fig. 7 Procedural terrain texturing

The final process for *building texturing* requires an algorithm to randomly identify material shaders and swap the textures based on different parts of the building. Using geometry normals to differentiate between different parts of a structure (roof, walls), various texture maps are applied to cover the façade (Figs. 6 and 7).

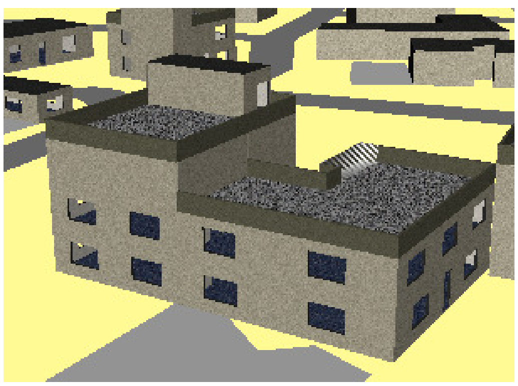

Fig. 8 Original COLLADA data

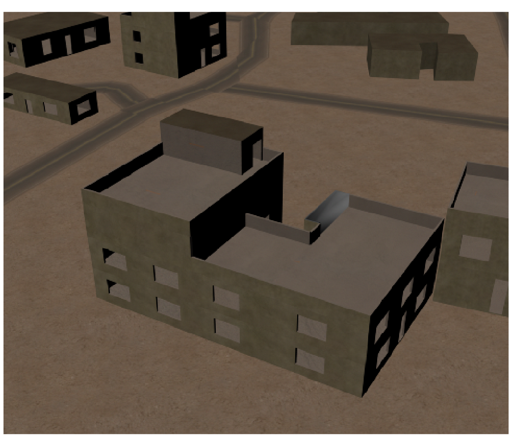

Fig. 9 Procedural terrain & building texturing

Once the data has been optimized and enhanced inside of Maya, there is the option to embed in it more abstract-level information that can be used by the artificial intelligence (AI) or human user within the game environment. Current modeling and simulation (M&S) environments typically rely on primitive elements of the terrain for an agent's decisions, and often at a very low-level such as used for path-planning and navigation. These elements do not contain level of fidelity required for representing complex and variable agent behavior such as culture. Geometry, collision surfaces, ground type, path nodes and pathing networks are well-suited for basic mobility and physics calculations but fail to accurately convey higher-level information that may be useful to an agent in achieving its goals. Our approach is to embed this contextual information (through annotations and affordances) directly in the virtual environment (i.e., terrain) and have the AI use this information in its decision making. Drawing upon other academic fields (psychology, sociology, business/management, healthcare, and security), a broad classification hierarchy of cultural characteristics has been developed that is derived from the types of models presented by researchers like Triandis (1989) and Hofstede (2005). Embedding this type of metadata in the virtual environment allows agents to apply context to the objects around them and, as a result, provide a more immersive and realistic simulation experience.

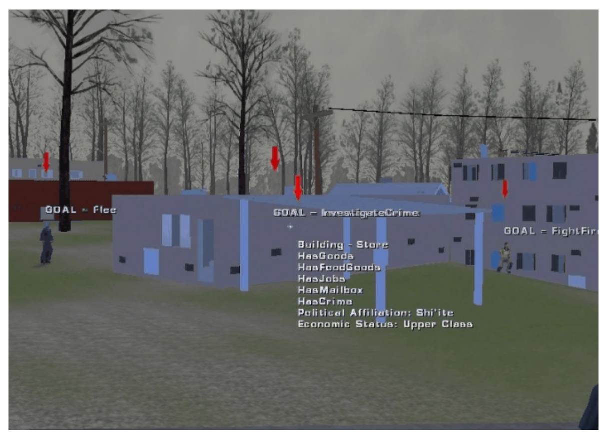

Fig. 10 An annotated building

After the data has been procedurally optimized and enhanced with metadata, it is then ready for export from Maya to one of several rendering environments such as Unreal, Gamebryo, OGRE, Delta3D, or VBS2. There is also the capability to export back to COLLADA for ingestion back into the RUGUD framework for creating correlated constructive simulation databases, such as OTFs. The export process is another manually intensive task because each game engine has specific requirements such as preferred image formats, LOD functionality, and proper use of physics meshes. The time required to complete such tasks range between hours to days to weeks depending on scene complexity. However, we have developed a set of automated exporters have reduced

the process to minutes. Employing the Maya exporters provided for each game engine, each part of the scene is exported to one of several native engine formats. These importers/exporters are procedurally called using MEL scripts, though they also utilize third-party software tools such as Feeling Software's COLLADA plugin.

5. CONCLUSIONS

To date, the MTGP has been tested and verified with several RUGUD datasets to create high-fidelity terrain for a variety of game technologies, including Gamebryo, Unreal Tournament, Delta3D and most recently Ogre and Half-Life 2. Analysis of the pipeline's manipulation and optimization process has shown an approximate 50% time savings for creating an area of game-ready terrain from existing source data (i.e., it would have taken a team of artists approximately double the amount of time to create the dataset had the MTGP not been used). The current procedural manipulations being done by the MTGP allow for a fully-automated export to one of the game engines listed above in a matter of minutes. Should the incoming source data (COLLADA) be deficient, or the manipulated Maya data be unsatisfactory in some way (e.g. not enough resolution), the option still exists for an artist to manually manipulate parts of the scene, which will have already been optimized in certain places.

Future work includes augmenting existing geo-specific datasets with representative environmental (geo-typical) features for an area of interest. Current geo-specific rendering of terrain requires information about both the geometry and photometry of an area. The geometry comprises the polygonal elevation map and above surface features that sit atop the terrain skin (vegetation, buildings, infrastructure). The photometry comprises the textured appearance of the geometry under certain conditions (lighting, atmospherics). However, in order for the modeled area to be truly accurate, a certain level of fidelity is required that ensures relevant features are included and represented correctly. For example, DTED-4 has a post spacing of 3 meters, which means the highest level of fidelity that can exist between vertices in the polygonal model is ~10 ft. This is often less than adequate for training the COE type of operations, which frequently require interactions with sub-meter environmental features (such as signage, utility poles, and narrow pathways). Unfortunately, even DTED-5 (1m) data for today's rendering systems are not adequate for a realistic representation of an area of interest. Additionally, much of the source data used does not contain the above surface features required for accurate training and simulation. Academia has investigated such feature placement and will be leveraged for future work (Greuter et al., 2003, Danaher, 2002). Candidate enhancement capabilities for inclusion in the MTGP include:

- Elevation alteration – the ability to create and modify the height map to produce terrain with varying degrees of height (Schneider et al., 2006)
- Feature/Structure placement – the ability to populate the terrain skin with structures and features that are representative of the area of interest (Prager et al., 2004)
- Texture application – the application of textures to the terrain skin and features/structures that are representative of the area of interest (Cohen et al., 2003)

As the simulation and training communities continue the shift towards procedural generation of game environments, RDECOM-STTC will direct appropriate enhancements for both RUGUD and U2MG. For example, RUGUD extensions are currently being performed for the Federal Law Enforcement Training Center (FLETC), which has built a simulation center to provide a multitude of simulation systems for law enforcement training purposes. Tasks involved with the FLETC effort include researching export capabilities for the SWAT4 and VBS2 gaming engines, as well as RUGUD pipeline improvements that facilitate ancillary tasks (texture assignment for building models, etc.).

Additionally , future game-related efforts for RUGUD and U2MG are likely to focus more on the commonalities between multiple game engines, rather than on specific formats. Research and implementation of industry-known optimizations for geometry, texturing, and other modeling aspects will be applied to the current pipeline infrastructure to enhance future data sets. This may include improvements to texture mapping, decaling, triangulation algorithms, levels of detail, and other methods meant to increase run-time performance. Improvements are also planned with respect to asset representation, such that we can derive an internal common model format structured to simplify the conversion process between SAF, visual, and gaming formats of interest. This will enable us to explore additional MTGP-related capabilities like the potential for re-ingesting modified COLLADA assets back into the RUGUD pipeline after they have been maniuplated by artists.

ACKNOWLEDGMENTS

The project or effort described here has been sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

REFERENCES

BBC, 2005: Cost of making games set to soar, *BBC Online,* November 17, 2005.

Campbell, C. Wertman, K. De la Cruz, J., 2006: A Framework for Generating High-Fidelity, Interoperable Urban Terrain Databases, *Proceedings of the Interservice/Industry Training, Simulation and Education Conference.*

Cohen, M., Shade, J., Hiller, S., and Deussen, O., 2003: Want Tiles for Image and Texture Generation, *ACM Transactions on Graphics.*

Epic Games, 2008: Unreal Engine 3, Unreal Wiki. http://wiki.beyondunreal.com/Legacy:Unreal_Engine_Versions/3.

Greuter, J., Parker, J., Stewart, N., and Leach, G., 2003: Real-time Procedural Generation of 'Pseudo-Infinite' Cities, *International Conference on Computer Graphics and Interactive Techniques,* Melbourne, Australia, February 11 – 14.

Introversion Software, 2008: Procedural Content Generation. http://www.gamecareerguide.com/features/336/procedural_content_.php?page=2

Mann, J., Eifert, L., 2006" Easing the Pain of Urban Modeling, *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*.

Prager, S., Cauble, K., Bakeman, D., Haes, S., and Goodman, G., 2004: Malls, Sprawl and Clutter: Realistic Terrain for Simulation of JUO, *The Interservice/Industry Training, Simulation & Education Conference,* Orlando, FL.

Procedural, Inc, 2008: CityEngine Main Website. http://www.procedural.com/.

Schneider, J., Boldte, T., and Westermann, R., 2006: Real-time Editing, Synthesis, and Rendering of Infinite Landscapes on GPUs. *Vision, Modeling, and Visualization*.

Stanford, 2004: The Reality of Video Games, *Stanford GSB News*, April 2004.

SECURITY CLASSIFICATION & PUBLIC INFORMATION RELEASE FORM
26th Army Science Conference

The purpose of this form is to document the security, OPSEC, and public information reviews of papers submitted for the 26th Army Science Conference (26th ASC) and to authorize presentation, videotaping, audiotaping and publication to CD/web (www) of papers selected for presentation at the 26th ASC, JW Marriott Orlando Grande Lakes, Orlando, FL, 1-4 December 2008. This form must accompany the final summary and manuscript. Without its completion, the summary and manuscript will not be considered for presentation or publication. Please complete all blocks and affix signatures.

TITLE OF PAPER: TOWARDS AN AUTOMATED PIPELINE FOR THE TRANSLATION & OPTIMIZATION OF GEOSPATIAL DATA VIRTUAL ENVIRONMENTS

NAME AND ORGANIZATION OF PRINCIPAL AUTHOR: RYAN McALINDEN

TELEPHONE: (Commercial Only): ____________________

PART I. SECURITY/OPSEC REVIEW - The final summary and manuscript must be reviewed by an authorized Security Officer IAW guidance under AR 380-5, DoD Directive 5200.20, AR 530-1, the Military Technologies List, and any other applicable security regulations. The paper must be unclassified and may not contain Restricted Data, Formerly Restricted Data or be marked NOFORN.

1. [ ] The summary is **UNCLASSIFIED/ PUBLIC DOMAIN**. (Only unclassified/ public domain summaries will be accepted.)

2. [ ] The final manuscript is **UNCLASSIFIED/ PUBLIC DOMAIN** and suitable for release and publication.

3. [✓] The manuscript and summary have undergone OPSEC review, and it has been determined that the material is suitable for release and publication in the public domain.

4. [ ] The proposed presentation has been reviewed to assure that classified information is not disclosed and that material is suitable for release to DoD attendees and cleared representatives of NATO/ ABCA countries.

5. Typed Name, Title, Grade, and Organization of OPSEC Reviewer: Mr. Angel Rodriguez. Div Chief
Signature of OPSEC Reviewer: [signature] 9/16/2008

6. Typed Name, Title, Grade, and Organization of Security Reviewer:
Signature of Security Reviewer: / 2008

PART II. INFORMATION RELEASE REVIEW. All papers must be reviewed for suitability of release (IAW AR 380-5), editorial accuracy, content and propriety. The reviewer must be knowledgeable of the research presented in the paper and cannot be an author or coauthor of said paper. No additional editing of this manuscript will be made prior to publication.

7. [✓] The summary and manuscript have been reviewed for conformance with release policy, editorial accuracy, content and propriety. They are suitable for presentation and publication in the public domain.

8. Typed Name, Title, Grade, and Organization of Information Release Reviewer: Michelle Millirven, Public Affairs Specialist, US ARMY RDECOM ~~STTC~~
Signature of Information Release Reviewer: for: [signature] 9/16/2008

Please fax completed form to 757-357-5108.

CONSENT TO PUBLISH AND TRANSFER OF COPYRIGHT

Under United States Copyright law, the transfer of copyright from the contributing author(s) should be explicitly stated in writing to enable the publisher to publish and disseminate the author's work to the fullest extent. The following agreement must be signed by each contributing author and returned with your manuscript. The form will be sent to the publisher should your work be selected for publication.

TITLE OF PAPER/ARTICLE: Towards an Automated Pipeline for the translation + Optimization of Geospatial Data for Virtual Environments

AUTHOR(S): Ryan McAlinden, Matt Bosack, Adrian Macha, Esau Vargas, John Mann, Tim Walker, Julio de la Cruz

PART A – COPYRIGHT TRANSFER FORM AND WARRANTIES

NOTE: Part A applies only to authors who are not U.S. government employees or whose submitted work is not a result of their official U.S. government duties. If the paper/article was prepared as part of the writer's duties for his/her non-U.S. government employer (a "work for hire"), this agreement must be signed by the employer as an Author. If the paper/article was prepared by a U.S. government employee as part of his/her official duties, please refer to Part B.

1. The undersigned author or authors ("Author") of the above paper/article ("Article") transfers and assigns exclusively to the selected publisher ("Publisher") all Author's right, title and interest in the Article including, without limitation, the copyright therein. These rights include without limitation, translation rights, mechanical, electronic and visual reproduction; electronic storage and retrieval; and all other forms of electronic publication or any other types of publication including subsidiary rights.

2. In return for said rights, the Publisher grants to the Author the following rights:

   a. All proprietary rights relating to the Article, other than copyright, such as patent rights.

   b. The right to use, after publication, part or all of the Article in subsequent works of the Author, provided written permission is granted by the Publisher and the proper acknowledgment is made to the source and to the publisher.

   c. The right to make oral presentation of the material in any form.

   d. In the case of work performed under United States Government contract, Publisher grants the U.S. Government royalty-free permission to reproduce all or portions of the Article and to authorize others to do so for U.S. Government purposes.

3. Any other use or reproducing of the work may require a fee and/or permission from the Publisher.

4. In the event the Article is not published, this agreement becomes null and void.

5. The Author warrants and represents that the Article is original work and has not been published before in any form and that it does not infringe upon any copyright or other right(s), that it does not contain infringing or other unlawful matter, that he or she is the sole and exclusive owner of the rights here-in conveyed to the Publisher, and that he or she has obtained the customary permission from the copyright owner or their legal representative whenever information from copyrighted material is quoted or a table or illustration from such material is used. The Author will indemnify the Publisher for, and hold the Publisher harmless from any loss, expense or damage occasioned by any claim or suit by a third party for copyright infringement or arising out of any breach of the foregoing warranties as a result of publication of the Article. The Article shall be delivered to the Publisher free of copyright charges.

______________________________ 9/22/08
AUTHOR(S) (and TITLE, if not author(s)) **DATE**

PART B – U.S. GOVERNMENT EMPLOYEE CERTIFICATION

NOTE: If your work was performed under a U.S. Government contract, but you are not a government employee, sign Part A of this form (above) and see Item 2d. of Part A.

This will certify that all authors of the above paper/article are employees of the U.S. Government and performed this work as part of their employment; that the paper/article is therefore not subject to U.S. copyright protection; and that no assignment of any copyright interest in the paper/article is made by this document. In the event that the paper/article is published in any foreign countries in which copyright protection is available under the laws of such countries, and Publisher acquires a copyright in the paper/article, Publisher grants to the U.S. Government a perpetual, paid-up, worldwide license to reproduce, copy, or otherwise use the paper/article in any manner.

[signature] 9/22/08

AUTHORIZED SIGNATURE/TITLE DATE

US ARMY RDECOM-STTC

NAME OF GOVERNMENT ORGANIZATION